

## GIỚI THIỆU MySQL

### Nội dung

1.Chuẩn bị .....	1
2.Khởi động và tắt MySQL Server.....	1
3.Quản trị MySQL với MySQL Administrator.....	1
4.Truy vấn với MySQL Query Browser.....	5
5.Kiểu dữ liệu trong MySQL.....	8
6.Tạo CSDL trong MySQL Administrator.....	13
7.Tạo Bảng trong MySQL Administrator.....	14
8.Tạo khóa chính .....	16
9.Tạo khóa ngoại .....	18
10.Insert dữ liệu vào bảng.....	23

### 1.Chuẩn bị

MySQL có hai phiên chính là MySQL Enterprise và MySQL Community Server. Trong phần thực hành này ta chọn MySQL Community Server phiên bản 5.0.45 và MySQL GUI Tools phiên bản 5.0.12 tại <http://dev.mysql.com/downloads>.

### 2.Khởi động và tắt MySQL Server

Để khởi động MySQL Server ta có thể dùng một trong hai cách sau:

a. MySQL Server sẽ tự động khởi động ngay sau khi khởi động Windows nếu trong quá trình cài đặt ta chọn thiết lập cài đặt MySQL như là một dịch vụ của Windows.

b. Sử dụng lệnh `mysqld-nt` như sau:

- Mở cửa sổ dòng lệnh Command Prompt bằng:
  - + Start → All Programs → Accessories → Command Prompt hoặc
  - + Start → Run và thực thi lệnh `cmd`.
- Từ cửa sổ dòng lệnh thực hiện lệnh `mysqld-nt [options]`  
Trong đó `[options]` là các tùy chọn khởi động cho MySQL Server. Để xem đầy đủ nội dung các tùy chọn ta sử dụng `mysqld-nt -verbose -help`. Nếu khởi động với các options mặc định thì chỉ dùng lệnh `mysqld-nt`, khi đó các options sẽ được lấy trong tập tin cấu hình `my.ini` trong thư mục cài đặt MySQL hoặc thư mục Windows của hệ thống.

Để tắt MySQL Server ta sử dụng lệnh `mysqladmin` như sau:

- Mở cửa sổ dòng lệnh Command Prompt.
- Từ cửa sổ dòng lệnh thực hiện lệnh `mysqladmin -u root shutdown`.

### 3.Quản trị MySQL với MySQL Administrator

Để dễ dàng tạo các cơ sở dữ liệu chúng ta có thể sử dụng công cụ hỗ trợ quản trị MySQL Server gọi là MySQL Administrator. Sau khi

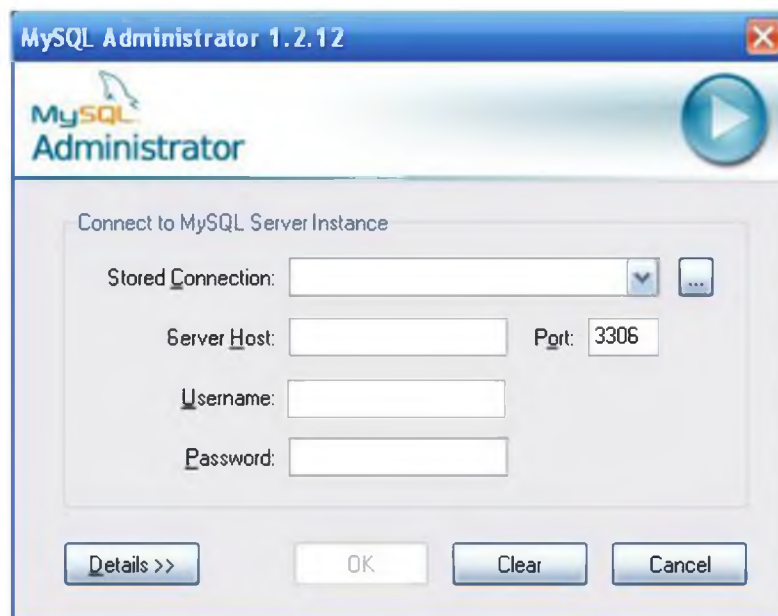
chắc chắn MySQL Server đã được khởi động, ta sẽ khởi động MySQL Administrator bằng cách chọn Start → All Programs → MySQL → MySQL Administrator. Cửa sổ đăng nhập MySQL Administrator xuất hiện (Hình 1.1).

Giả sử MySQL Server được cài đặt trên máy đơn mà ta đang sử dụng. Để kết nối với MySQL Server ta cần các thông số sau cơ bản sau:

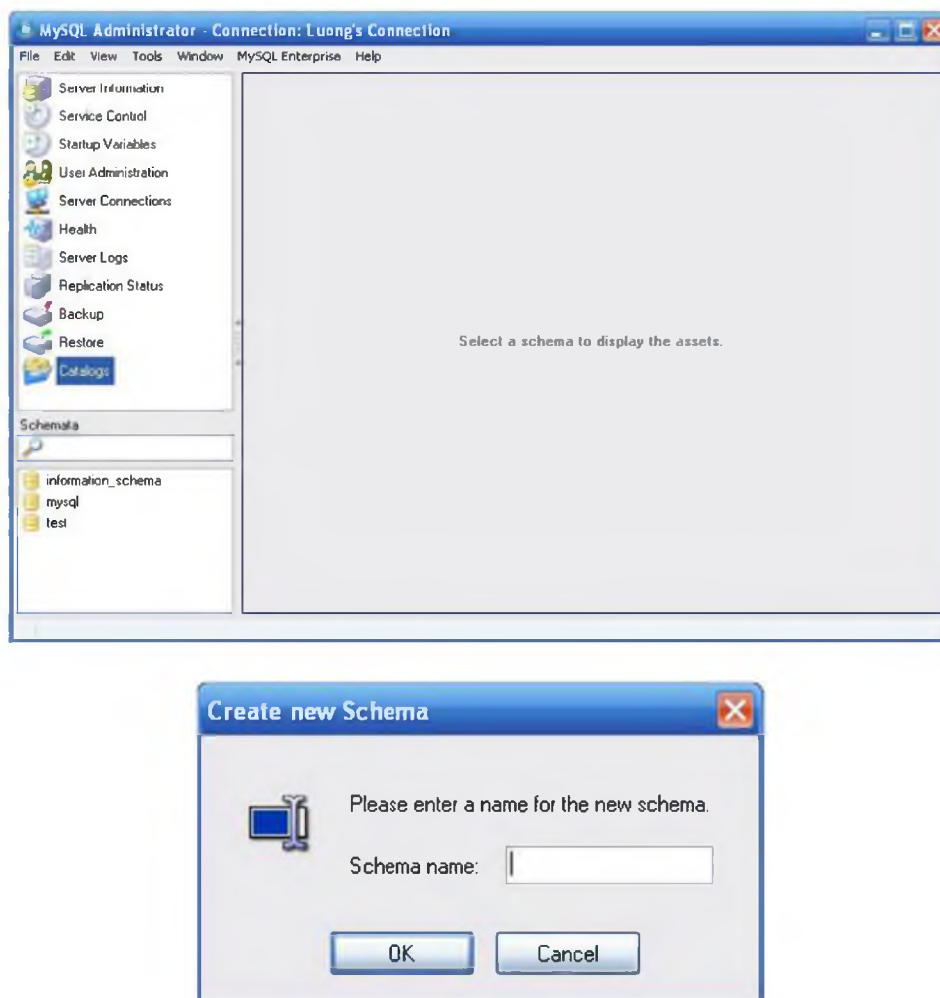
- Server Host: localhost
- Port: 3306
- Username: root
- Password:

Sau đó nhấp chuột vào nút OK. Nếu kết nối thành công của sổ chính của MySQL Administrator sẽ xuất hiện (Hình 1.2).

Để tạo cơ sở dữ liệu ta chọn mục Catalogs trong phần bên trái của cửa sổ chính. Nhấp chuột phải vào phần ở góc trái dưới của cửa sổ chính và chọn Create New Schema, chú ý trong MySQL một cơ sở dữ liệu còn được gọi là Schema. Hộp thoại Create New Schema xuất hiện (Hình 1.3) và yêu cầu ta nhập vào Schema name, tên của cơ sở dữ liệu cần tạo.



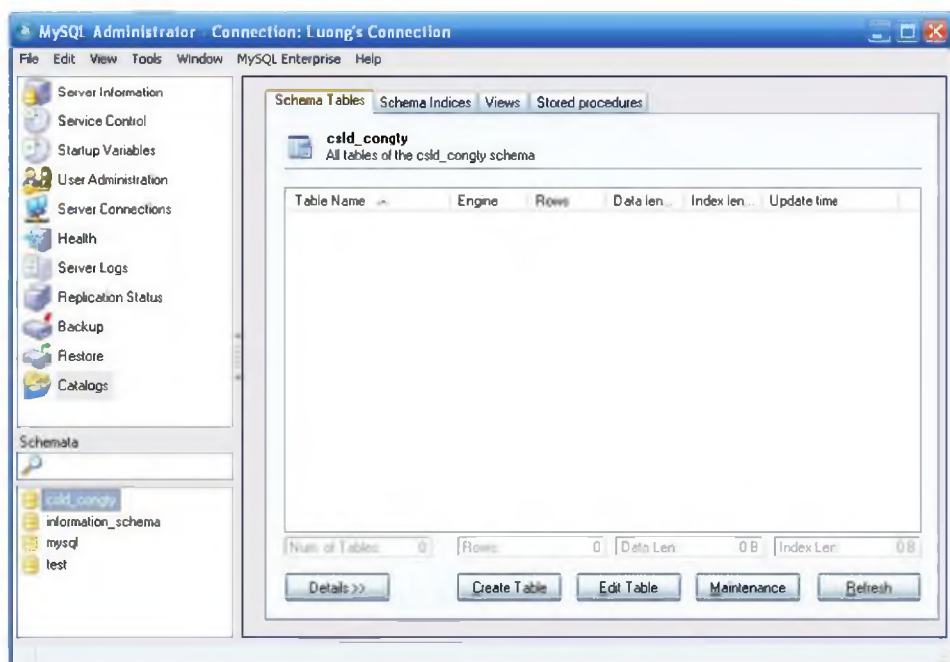
Hình 1.1: Cửa sổ đăng nhập MySQL Administrator



**Hình 1.3: Hộp thoại Create New Schema**

Giả sử ta tạo một cơ sở dữ liệu có tên là CSDL\_Congty, khi đó Schema csdl\_congty sẽ xuất hiện trong danh sách các Schema ở góc trái dưới của cửa sổ chính.

Để tạo mới, chỉnh sửa, xóa các quan hệ (bảng) của một cơ sở dữ liệu, ta nhấp chuột để chọn cơ sở dữ liệu (schema) trong danh sách schema. Trong phần bên phải của cửa sổ chính sẽ xuất hiện mục Schema Tables cho phép thực hiện việc tạo, hiệu chỉnh và xóa quan hệ (Hình 1.4).



Hình 1.4: Cửa sổ chính với mục Schema Tables

**Reset mật khẩu của root trên Windows:**

vào Start -> Run -> gõ

Code:

services.msc

Enter

Tìm services MySQL và stop lại

sau đó tìm thư mục cài đặt MySQL giả sử C:\mysql

Vào Start-> Run -> gõ

Code:

cmd

Enter.

ta vào thư mục bin

Code:

cd C:\mysql\bin

và gõ lệnh

Code:

C:\mysql\bin>mysqld-nt --skip-grant-tables

Giữ nguyên cửa sổ đó và mở thêm 1 cửa sổ command mới

vào bin và gõ lệnh

Code:

C:\mysql\bin>mysql -u root

khi đó sẽ login vào root mà không cần password.

Để Reset password root ta dùng lệnh

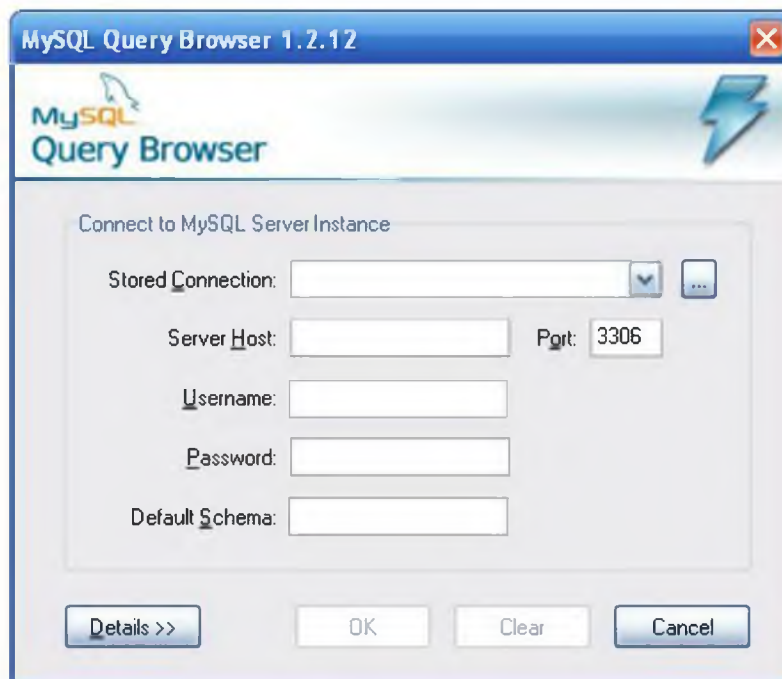
Code:

```
mysql>use mysql; update user set Password=PASSWORD('new_pass') WHERE  
User='root';FLUSH PRIVILEGES;
```

trong đó new\_pass chính là password của bạn.

#### 4. Truy vấn với MySQL Query Browser

MySQL Query Browser cho phép ta thực hiện các lệnh truy vấn cơ sở dữ liệu. Để sử dụng MySQL Query Browser ta khởi động nó từ Start → All Programs → MySQL → MySQL Query Browser. Khi đó cửa sổ đăng nhập MySQL Query Browser xuất hiện (Hình 1.5).



Hình 1.5: Cửa sổ đăng nhập MySQL Query Browser

Giả sử MySQL Server được cài đặt trên máy đơn mà ta đang sử dụng. Ta cũng nhập vào các thông số cơ bản tương tự khi kết nối MySQL Administrator với MySQL Server:

- Server Host: localhost
- Username: root
- Password:

Trong mục Default Schema phải nhập vào tên cơ sở dữ liệu mà ta muốn thực hiện các truy vấn trên đó. Sau đó nhấp chuột vào nút OK và cửa sổ chính của MySQL Query Browser sẽ xuất hiện (Hình 1.6).

Cửa sổ chính của MySQL Query Browser gồm các thành phần cơ bản sau.

##### c. Công cụ truy vấn - Query Toolbar

Nơi để soạn thảo và thực thi các truy vấn. Gồm có 3 nút di chuyển (Go back, Next, Refresh), vùng soạn thảo truy vấn, 2 nút chức năng (Execute, Stop).

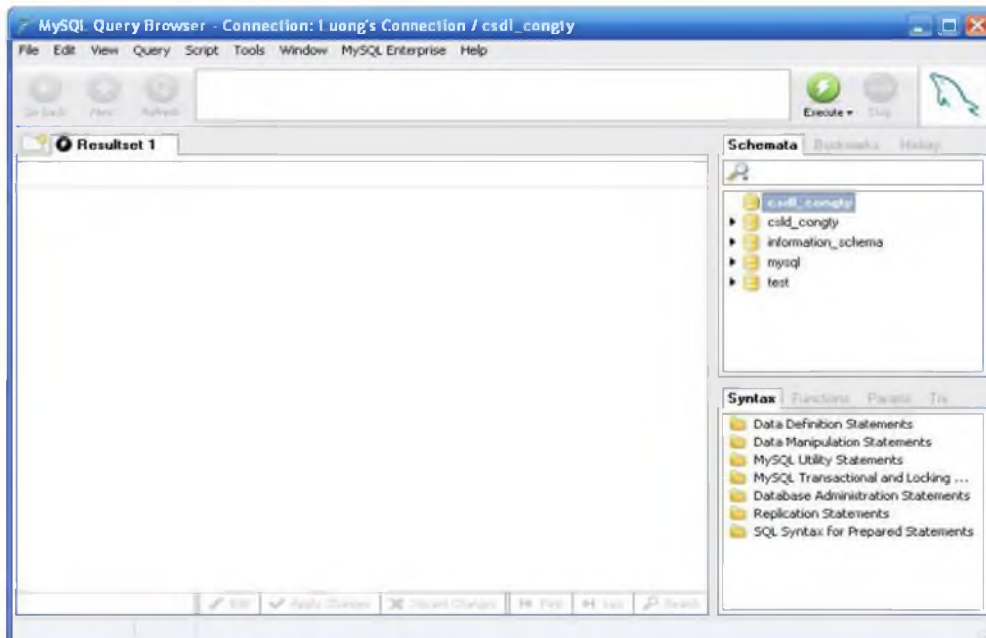
- Các nút di chuyển

Dùng để duyệt danh mục các lệnh truy vấn đã được thực thi. Chỉ các lệnh truy vấn không xảy ra lỗi mới được lưu vào danh mục. Nút *Go back* chuyển về nhóm lệnh truy vấn đã thực thi

ngay trước nhóm lệnh hiện đang có trong vùng soạn thảo truy vấn. Nút *Next* chuyển đến nhóm lệnh truy vấn đã thực thi ngay sau nhóm lệnh hiện đang có trong vùng soạn thảo truy vấn. Nút *Refresh* thực thi lại nhóm lệnh truy vấn đã được thực thi gần nhất.

- Vùng soạn thảo truy vấn  
Dùng hiển thị và soạn thảo các lệnh truy vấn. Vùng này mặc định có 3 dòng và có thể được mở rộng lên tối đa 10 dòng. Nếu các truy vấn dài hơn 10 dòng thì thanh cuộn sẽ xuất hiện. Để tăng tối đa kích thước vùng soạn thảo dùng View → Maximize Query Edit hoặc gõ phím F11.
- Các nút chức năng  
Dùng để thực thi hoặc dùng các lệnh truy vấn. Nút *Execute* dùng để thực thi các lệnh truy vấn hiện thời trong vùng soạn thảo. Nút *Stop* dùng để chấm dứt các lệnh thực thi nếu việc thực thi xảy ra quá lâu mà chưa cho ra kết quả.

Để lưu lại các câu lệnh truy vấn đã thực thi vào tập tin dùng File → Save As. Sau đó ta chọn kiểu tập tin để lưu là SQL Script File ANSI (\*.sql).



Hình 1.6: Cửa sổ chính của MySQL Query Browser

#### d. Công cụ truy vấn - Query Toolbar

Nơi để soạn thảo và thực thi các truy vấn. Gồm có 3 nút di chuyển (Go back, Next, Refresh), vùng soạn thảo truy vấn, 2 nút chức năng (Execute, Stop).



- Các nút di chuyển  
Dùng để duyệt danh mục các lệnh truy vấn đã được thực thi. Chỉ các lệnh truy vấn không xảy ra lỗi mới được lưu vào danh mục. Nút *Go back* chuyển về nhóm lệnh truy vấn đã thực thi ngay trước nhóm lệnh hiện đang có trong vùng soạn thảo truy vấn. Nút *Next* chuyển đến nhóm lệnh truy vấn đã thực thi ngay sau nhóm lệnh hiện đang có trong vùng soạn thảo truy vấn. Nút *Refresh* thực thi lại nhóm lệnh truy vấn đã được thực thi gần nhất.
- Vùng soạn thảo truy vấn  
Dùng hiển thị và soạn thảo các lệnh truy vấn. Vùng này mặc định có 3 dòng và có thể được mở rộng lên tối đa 10 dòng. Nếu các truy vấn dài hơn 10 dòng thì thanh cuộn sẽ xuất hiện. Để tăng tối đa kích thước vùng soạn thảo dùng View → Maximize Query Edit hoặc gõ phím F11.
- Các nút chức năng  
Dùng để thực thi hoặc dùng các lệnh truy vấn. Nút *Execute* dùng để thực thi các lệnh truy vấn hiện thời trong vùng soạn thảo. Nút *Stop* dùng để chấm dứt các lệnh thực thi nếu việc thực thi xảy ra quá lâu mà chưa cho ra kết quả.

Để lưu lại các câu lệnh truy vấn đã thực thi vào tập tin dùng File → Save As. Sau đó ta chọn kiểu tập tin để lưu là SQL Script File ANSI (\*.sql).

e. *Vùng hiển thị kết quả truy vấn - Result Area*

Nơi hiển thị kết quả của các lệnh truy vấn.

f. *Vùng duyệt đối tượng - Object Browser*

Nơi liệt kê và cho phép lựa chọn cơ sở dữ liệu, quản lý bookmark và danh mục các câu lệnh truy vấn đã được thực hiện.

Trình duyệt cơ sở dữ liệu, thẻ Schemata, là thành phần chính của vùng duyệt đối tượng. Ta có thể chọn các quan hệ và các thuộc tính của quan hệ, có thể hiệu chỉnh, hủy và tạo bảng, có thể hủy và tạo cơ sở dữ liệu. Ta cũng có thể lựa chọn cơ sở dữ liệu mặc định để thực hiện các truy vấn trên cơ sở dữ liệu đó.

Cơ sở dữ liệu mặc định luôn được tô đậm, để thay đổi cơ sở dữ liệu mặc định nhấp chuột phải vào cơ sở dữ liệu muốn đổi và chọn Make Default Schema.

g. *Vùng duyệt thông tin - Information Browser*

Nơi cho phép tham khảo nhanh các thông tin về cú pháp các lệnh, cách sử dụng các toán tử và các hàm có sẵn trong MySQL.

h. *Vùng soạn thảo script - Script Editor*

Nơi để soạn thảo các nhóm lệnh truy vấn lớn. Vùng này nằm trong một thẻ riêng biệt trong vùng hiển thị kết quả truy vấn. Khi vùng này xuất hiện, công cụ Advanced Toolbar xuất hiện và thay thế Query Toolbar.

Để tạo vùng soạn thảo mới dùng File → New Script Tab. Để lưu lại nội dung trong vùng soạn thảo dùng File → Save As và chọn kiểu tập tin để lưu là SQL Script File ANSI (\*.sql).

Các nút chức năng gỡ lỗi của công cụ Advanced Toolbar có thể sử dụng với Script Editor gồm:

- *Execute*: thực thi tất cả các lệnh truy vấn trong vùng soạn thảo.
- *Continue*: thực thi lệnh truy vấn đầu tiên (hoặc lệnh truy vấn tại vị trí con trỏ) và kết thúc khi xảy ra lỗi hoặc gặp một điểm gãy (break point).
- *Step*: thực thi lệnh truy vấn kế tiếp.
- *Pause*: dừng việc thực thi và tô đậm lệnh sẽ được thực thi kế tiếp.
- *Stop*: dừng việc thực thi.

## 5. Kiểu dữ liệu trong MySQL

### i. Các kiểu số

MySQL hỗ trợ tất cả các kiểu số của SQL chuẩn gồm các kiểu số chính xác (INTEGER, SMALLINT, DECIMAL và NUMERIC) và các kiểu số gần đúng (FLOAT, REAL và DOUBLE PRECISION). INT là viết tắt của INTEGER và DEC là viết tắt của DECIMAL.

Ngoài ra, MySQL cũng hỗ trợ các kiểu số của SQL mở rộng là TINYINT, MEDIUMINT và BIGINT. Bảng 1.1 liệt kê kích thước và phạm vi của các kiểu số.

Kiểu	Kích thước (Bytes)	Giá trị nhỏ nhất (Dấu / Không dấu)	Giá trị lớn nhất (Dấu / không dấu)
TINYINT	1	-128	127
		0	255
SMALLINT	2	-32,768	32,767
		0	65,535
MEDIUMINT	3	-8,388,608	8,388,607
		0	16,777,215
INT	4	-2,147,483,648	2,147,483,647
		0	4,294,967,295
BIGINT	8	-9,223,372,036,854,775	9,223,372,036,854,775,807



		,808	
		0	18,446,744,073,709,5 51,615

Bảng 1.1: Kích thước và phạm vi của các kiểu số

Với kiểu số gần đúng MySQL dùng 4 bytes biểu diễn giá trị gần đúng và 8 bytes biểu diễn giá trị chính xác. Kiểu FLOAT và DOUBLE dùng để biểu diễn kiểu số gần đúng. Kiểu số gần đúng với độ chính xác từ 0 - 23 chữ số được biểu diễn bằng kiểu FLOAT và độ chính xác từ 24 - 53 chữ số được biểu diễn bằng kiểu DOUBLE.

MySQL cũng cho phép biểu diễn độ chính xác của kiểu gần đúng với cú pháp `FLOAT(M, D)`, `REAL(M, D)` và `DOUBLE PRECISION(M, D)` với M là tổng số chữ số và D là số chữ số phần thập phân. Ví dụ, `FLOAT(7, 4)` biểu diễn số chữ số có dạng -999.9999. MySQL cũng làm tròn số khi lưu trữ. Ví dụ, nếu giá trị là 999.00009 với kiểu `FLOAT(7, 4)` sẽ được làm tròn thành 999.0001.

Kiểu `DOUBLE PRECISION` trong SQL mở rộng cũng tương đương với kiểu `DOUBLE`.

#### j. Kiểu thời gian

Các kiểu ngày tháng và giờ dùng để biểu diễn các giá trị thời gian là `DATETIME`, `DATE`, `TIMESTAMP`, `TIME` và `YEAR`. Mỗi kiểu thời gian có một vùng giá trị hợp lệ và một giá trị gọi là zero dùng để chỉ một giá trị không hợp lệ mà MySQL không thể biểu diễn. Kích thước của các kiểu thời gian được cho trong Bảng 1.2.

Kiểu	Kích thước (Bytes)
DATE	3
TIME	3
DATETIME	8
TIMESTAMP	4
YEAR	1

Bảng 1.2: Kích thước của các kiểu thời gian

Các chú ý khi sử dụng các kiểu thời gian:

- MySQL khôi phục các giá trị của kiểu ngày hoặc giờ theo một định dạng xuất chuẩn, nhưng có thể hiểu nhiều định dạng cho các giá trị nhập.
- Các kiểu ngày tháng với giá trị có năm là hai chữ số gây ra

sự nhập nhằng vì thành phần thiên niên kỷ không được xác định cụ thể. MySQL hiểu các giá trị có năm là hai chữ số theo quy tắc sau:

- + Giá trị của năm từ 70 - 99 được hiểu là từ 1970 - 1999.
- + Giá trị của năm từ 00 - 69 được hiểu là từ 2000 - 2069.

- Mặc dù MySQL có thể hiểu các giá trị theo nhiều định dạng, nhưng giá trị của các kiểu ngày tháng phải luôn được nhập vào theo thứ tự năm-tháng-ngày. Ví dụ, 98-09-04 là ngày 04 tháng 09 năm 1998.
- MySQL tự động chuyển một giá trị kiểu ngày tháng hoặc giờ thành một số nếu giá trị được dùng ở dạng số và ngược lại.
- Khi MySQL gặp phải một giá trị kiểu ngày tháng hoặc giờ nằm ngoài phạm vi biểu diễn hoặc không hợp lệ, nó sẽ chuyển giá trị này thành giá trị zero ứng với kiểu tương ứng. Riêng giá trị nằm ngoài phạm vi của kiểu TIME sẽ được cắt để thành một giá trị giới hạn thích hợp của kiểu TIME.
- Zero là các giá trị đặc biệt, nhưng ta có thể lưu hoặc xem chúng một cách tường minh như các giá trị trong bảng bên dưới. Ta cũng có thể viết một cách đơn giản giá trị zero là 0.

Kiểu	Giá trị zero
DATETIME	'0000-00-00 00:00:00'
DATE	'0000-00-00'
TIMESTAMP	'0000-00-00 00:00:00'
TIME	'00:00:00'
YEAR	0000

Bảng 1.3: Giá trị zero của các kiểu

- Kiểu DATETIME, DATE và TIMESTAMP  
Kiểu DATETIME được dùng khi ta cần biểu diễn các giá trị chứa cả thông tin ngày tháng và giờ. MySQL khôi phục và hiển thị giá trị DATETIME theo định dạng 'YYYY-MM-DD HH:MM:SS'. Phạm vi biểu diễn từ '1000-01-01 00:00:00' đến '9999-12-31 23:59:59'.

Kiểu DATE được dùng khi ta cần biểu diễn các giá trị chỉ chứa thông tin ngày tháng. MySQL khôi phục và hiển thị giá trị DATE theo định dạng 'YYYY-MM-DD'. Phạm vi biểu diễn từ '1000-01-01' đến '9999-12-31'.

Kiểu `TIMESTAMP` có nhiều tính chất phụ thuộc vào phiên bản MySQL và chế độ SQL mà hệ quản trị đang chạy.

Có nhiều định dạng để định nghĩa các giá trị `DATETIME`, `TIME` và `TIMESTAMP`:

- Chuỗi có dạng `'YYYY-MM-DD HH:MM:SS'` hoặc `'YYYY-MM-DD'`. Ví dụ, `'1998-12-31 11:30:45'` là 11 giờ 30 phút 45 giây ngày 31 tháng 12 năm 1998
- Chuỗi có dạng `'YYYYMMDDHHMMSS'` hoặc `'YYYYMMDD'`. Ví dụ `'19981231113045'` là 11 giờ 30 phút 45 giây ngày 31 tháng 12 năm 1998.
- Số có dạng `YYYYMMDDHHMMSS` hoặc `YYYYMMDD`. Ví dụ `19830905132800` là 13 giờ 28 phút ngày 05 tháng 09 năm 1983. Các giá trị không hợp lệ được chuyển thành zero của kiểu tương ứng.

Khi MySQL Server chạy với chế độ `MAXDB` thì kiểu `TIMESTAMP` và `DATETIME` là như nhau. Khi không chạy với chế độ `MAXDB` thì các thuộc tính có kiểu `TIMESTAMP` có giá trị mặc định là `NOT NULL`.

- Kiểu `TIME`

MySQL khôi phục và hiển thị các giá trị `TIME` theo định dạng `'HH:MM:SS'` (hoặc định dạng `'HHH:MM:SS'` với các giá trị giờ lớn). Các giá trị `TIME` có phạm vi biểu diễn từ `'-838:59:59'` đến `'838:59:59'`. Thành phần giờ có thể có giá trị lớn vì kiểu `TIME` không chỉ được dùng để biểu diễn giờ trong một ngày mà còn được dùng để biểu diễn khoảng thời gian giữa hai mốc sự kiện.

Có nhiều định dạng để định nghĩa các giá trị `TIME`:

- Chuỗi có dạng `'D HH:MM:SS'`.
- Chuỗi có dạng `'HHMMSS'` nếu giá trị biểu diễn giờ trong một ngày. Ví dụ, `'101112'` là 10 giờ 11 phút 12 giây.
- Số có dạng `HHMMSS` nếu giá trị biểu diễn giờ trong một ngày. Ví dụ, `101112` là 10 giờ 11 phút 12 giây.

Các giá trị nằm ngoài phạm vi biểu diễn của kiểu `TIME` sẽ được chuyển về giá trị giới hạn thích hợp. Ví dụ, `'-850:00:00'` trở thành `'-838:59:59'`. Các giá trị không hợp lệ trở thành `'00:00:00'`. Vậy khi gặp một giá trị `'00:00:00'` ta không thể nói rằng nó là một giá trị được định nghĩa bởi `'00:00:00'` hay là một giá trị không hợp lệ.

- Kiểu `YEAR`

Kiểu `YEAR` dùng để biểu diễn các giá trị năm. MySQL khôi phục và hiển thị các giá trị `YEAR` theo định dạng `YYYY`. Phạm vi biểu diễn từ 1901 đến 2155.

Có nhiều định dạng để định nghĩa các giá trị `YEAR`:

- Chuỗi 4 chữ số từ '1901' đến '2155'. Ví dụ, '1990' là năm 1990.
- Số 4 chữ số từ 1901 đến 2155. Ví dụ, 1990 là năm 1990.
- Chuỗi 2 chữ số từ '00' đến '99'. Ví dụ, '90' là năm 1990. Lưu ý sự nhập nhằng khi thiếu thành phần thiên niên kỷ.
- Số từ 1 đến 99. Ví dụ, 50 là năm 2050. Lưu ý sự nhập nhằng khi thiếu thành phần thiên niên kỷ. Trong trường hợp này ta không thể sử dụng số 0 để chỉ là năm 2000 mà ta phải dùng chuỗi '0' hoặc '00' vì số 0 được hiểu là giá trị zero. Các giá trị không hợp lệ được chuyển thành zero.

- Vấn đề năm 2000 và các kiểu thời gian  
MySQL Server không gặp vấn đề với năm 2000 (Y2K).

*k. Kiểu chuỗi*

Các kiểu chuỗi gồm CHAR, VARCHAR, BINARY, VARBINARY, BLOB, TEXT, ENUM.

- Kiểu CHAR và VARCHAR  
Kiểu CHAR và VARCHAR tương tự nhau nhưng khác nhau cách các giá trị của chúng được lưu trữ và khôi phục.

Kiểu CHAR và VARCHAR được khai báo với độ dài để định rõ số tối đa các ký tự mà một giá trị có thể có. Ví dụ, CHAR(30) thì giá trị là một chuỗi có thể chứa tối đa 30 ký tự.

Với kiểu CHAR độ dài có thể có giá trị từ 0 đến 255. Khi một chuỗi giá trị kiểu CHAR được lưu, các ký tự trắng được ghép vào cuối để chuỗi có chiều dài như chiều dài đã định. Khi một chuỗi giá trị CHAR được khôi phục, các ký tự trắng ở cuối chuỗi sẽ bị xóa.

Các giá trị kiểu VARCHAR là các chuỗi có độ dài thay đổi. Độ dài có thể có giá trị từ 0 đến 65535. Khác với kiểu CHAR, các giá trị kiểu VARCHAR được lưu với độ dài thực.

Nếu các chuỗi giá trị có độ dài lớn hơn độ dài đã được xác định thì chuỗi sẽ bị cắt bớt các ký tự ở cuối để có được độ dài như đã xác định. Bảng 1.4 minh họa sự khác nhau của kiểu CHAR và VARCHAR.

Chuỗi giá trị	CHAR(4)	VARCHAR(4)
' '	' '	' '
'ab'	'ab '	'ab'
'abcd'	'abcd'	'abcd'
'abcdefg'	'abcd'	'abcd'

Bảng 1.4: Sự khác nhau của kiểu CHAR và VARCHAR

- Kiểu BINARY và VARBINARY

Kiểu BINARY và VARBINARY tương tự CHAR và VARCHAR nhưng các giá trị của nó là các chuỗi nhị phân, đó là các chuỗi byte. Do đó các giá trị này có thể sắp xếp và so sánh dựa trên các giá trị số của các byte.

Khái niệm độ dài của BINARY và VARBINARY cũng tương tự CHAR và VARCHAR nhưng độ dài được tính theo số byte thay vì tính theo số ký tự. Ví dụ, BINARY(5) thì giá trị là một chuỗi nhị phân 5 byte.

- Kiểu BLOB và TEXT

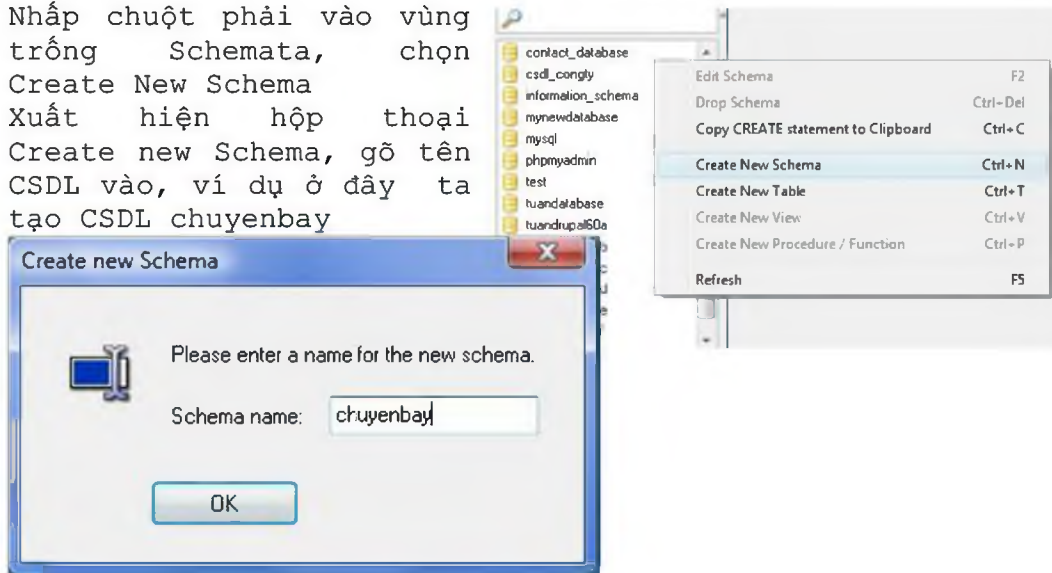
Kiểu BLOB và TEXT chứa các giá trị là chuỗi nhị phân và chuỗi ký tự lớn.

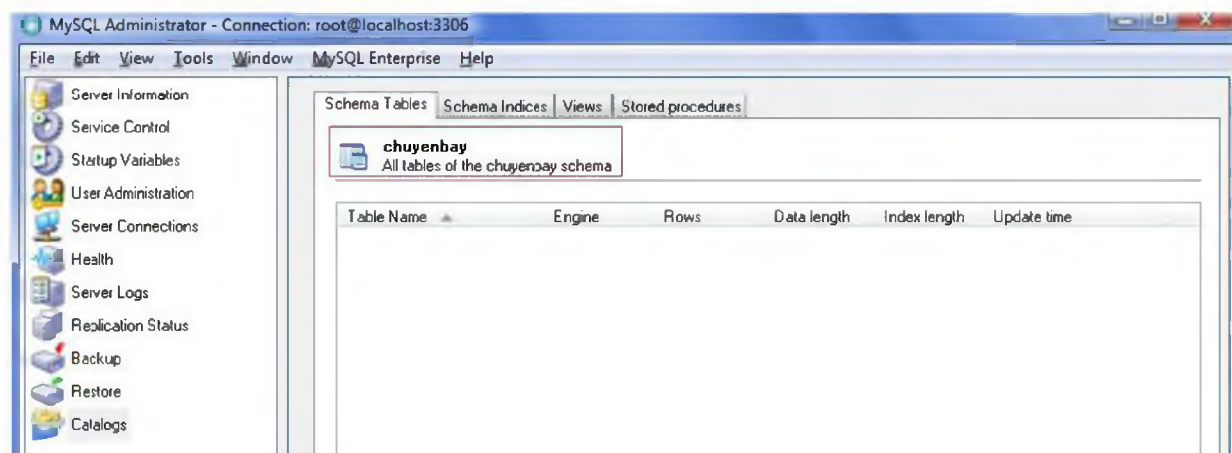
- Kiểu ENUM

Kiểu ENUM chứa các giá trị là các chuỗi được liệt kê trước.

## 6. Tạo CSDL trong MySQL Administrator

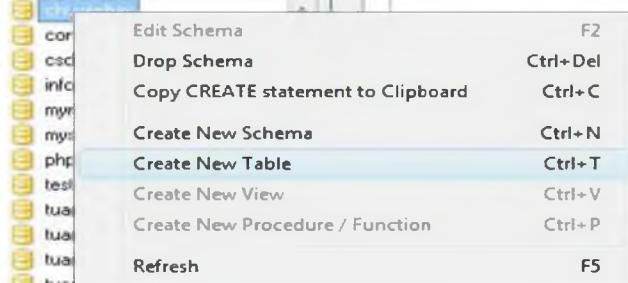
- Nhấp chuột phải vào vùng trống Schemata, chọn Create New Schema
- Xuất hiện hộp thoại Create new Schema, gõ tên CSDL vào, ví dụ ở đây ta tạo CSDL chuyenbay

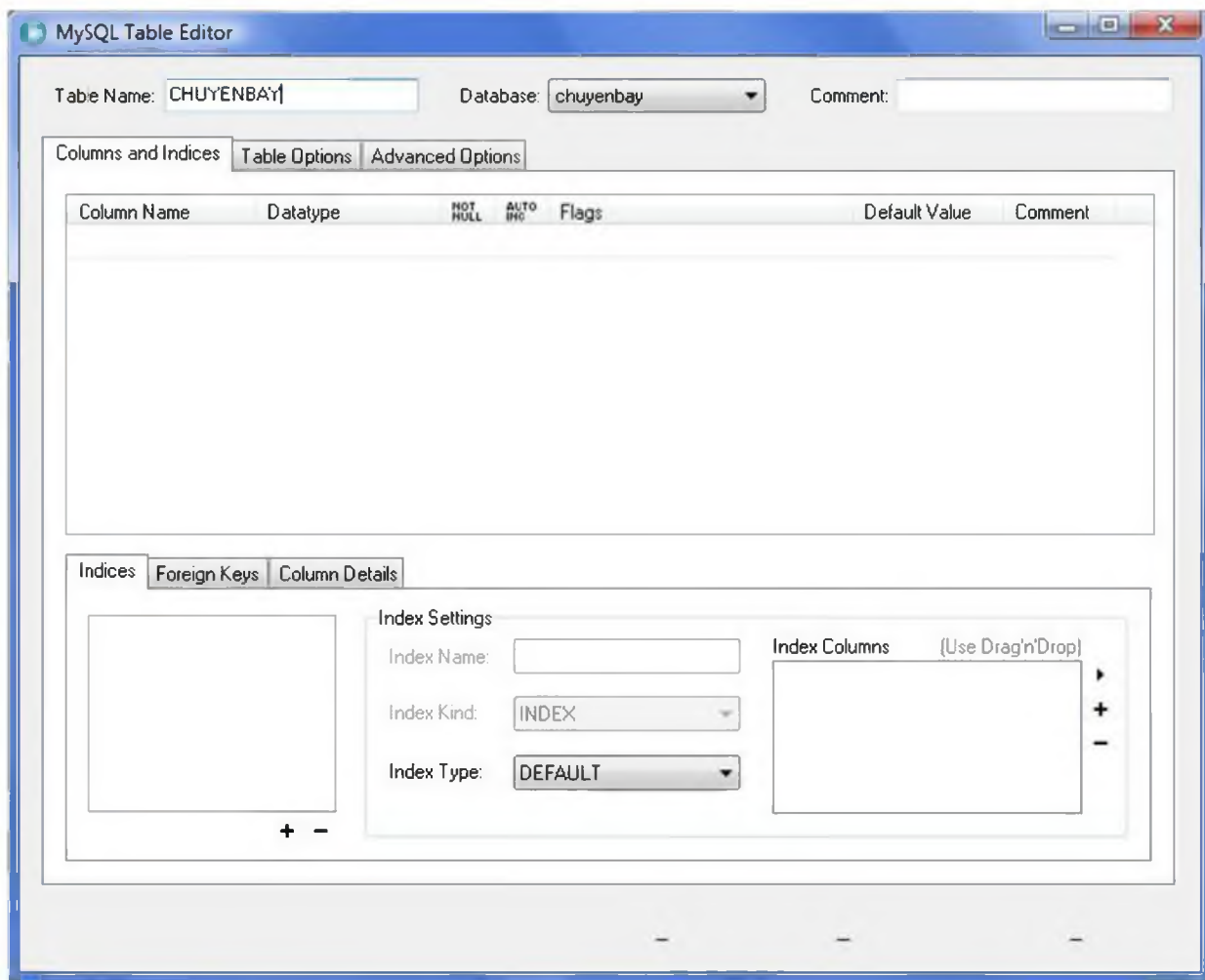




## 7. Tạo Bảng trong MySQL Administrator

- Nhấp chuột phải vào CSDL cần tạo bảng trong vùng Schemata. Ví dụ ta chọn CSDL chuyenbay để tạo bảng như hình bên.
- Chọn Create New Table
- Xuất hiện hộp thoại MySQL Table Editor như bên dưới





+Vùng tab gồm 3 phần:

- *Columns and Indices*: sử dụng tab này để tạo và hiệu chỉnh thông tin chỉ số và cột của bảng. Ta cũng tạo quan hệ KHÓA NGOẠI trong tab này.
- *Table Options*: Ta sử dụng tab Table Options để chọn storage engine và bảng mã ký tự cho bảng đó
- *Advanced Options*: Tab này để cấu hình một số tùy chọn

+ Trong vùng soạn thảo cột, ta có thể hiệu chỉnh tên, loại dữ liệu, giá trị mặc định và các thuộc tính khác cho cột.

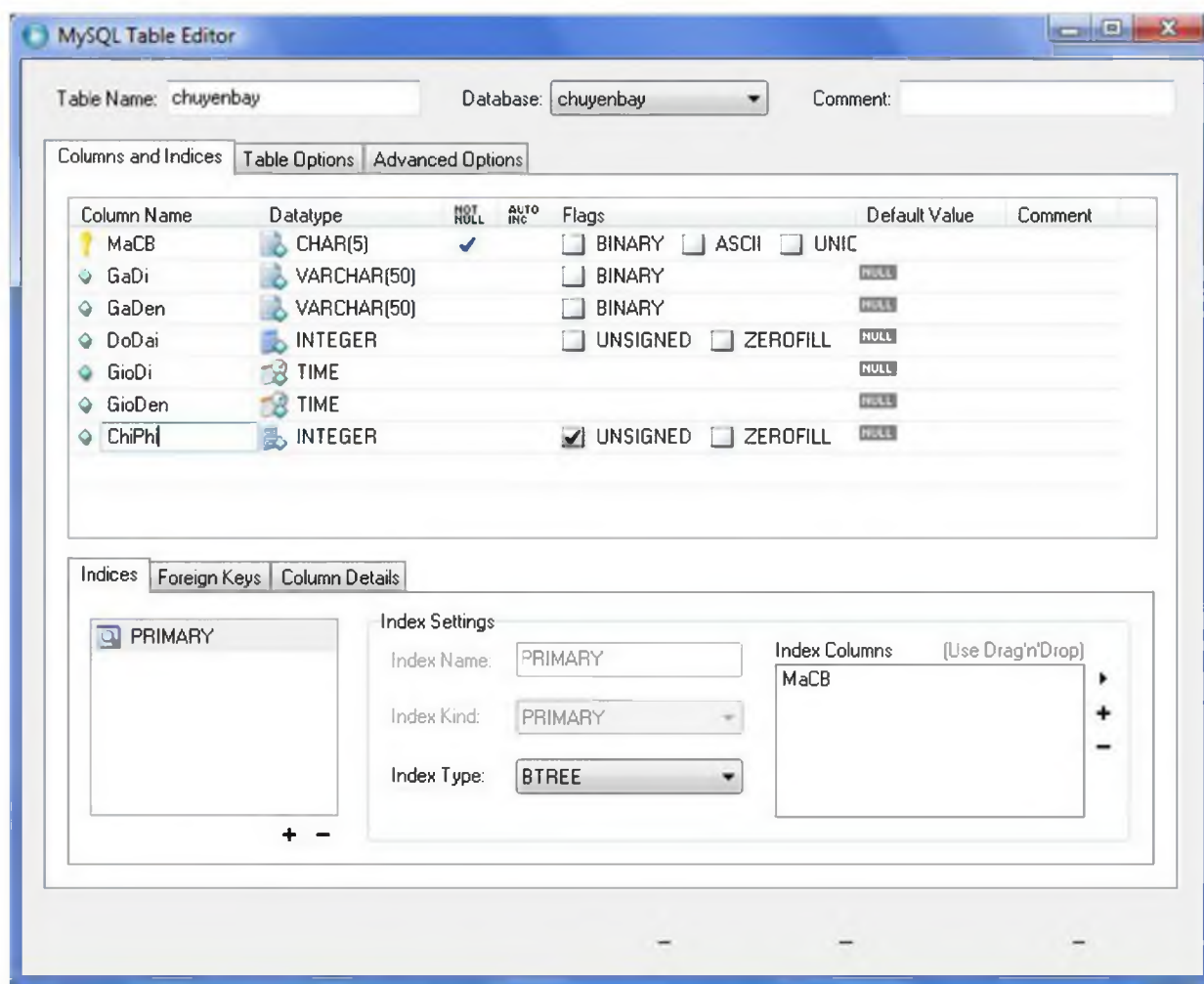
**- Tạo bảng:**

+ Ta gõ tên bảng vào ô Table Name

**- Tạo field cho bảng:**

+ Ta gõ tên field trong cột Column Name, kiểu dữ liệu của field trong cột Datatype. Nếu field đó bắt buộc phải có giá trị thì check vào cột NOT NULL,... Ví dụ ta tạo các field cho bảng **chuyenbay** trong CSDL **chuyenbay** như hình bên dưới.

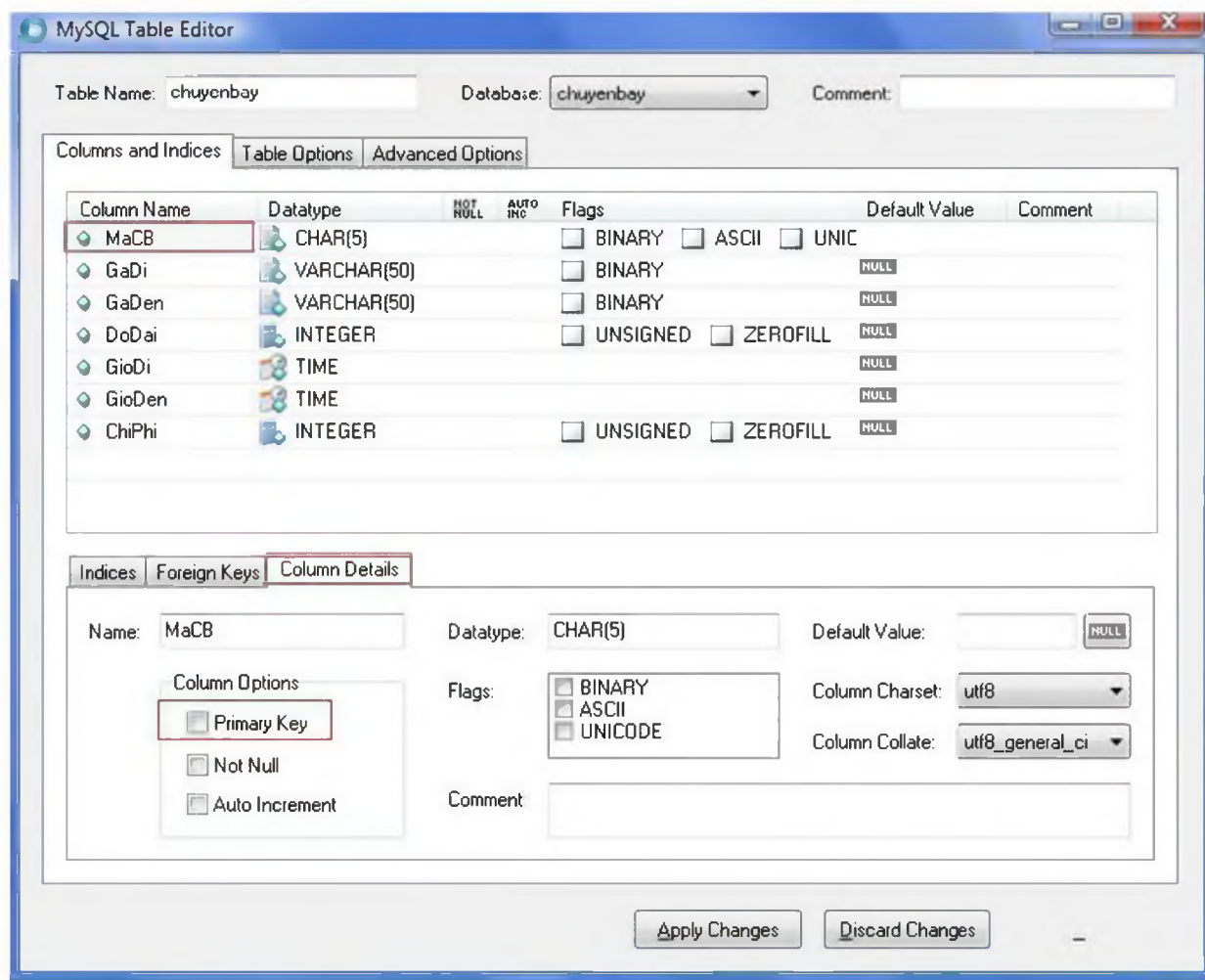




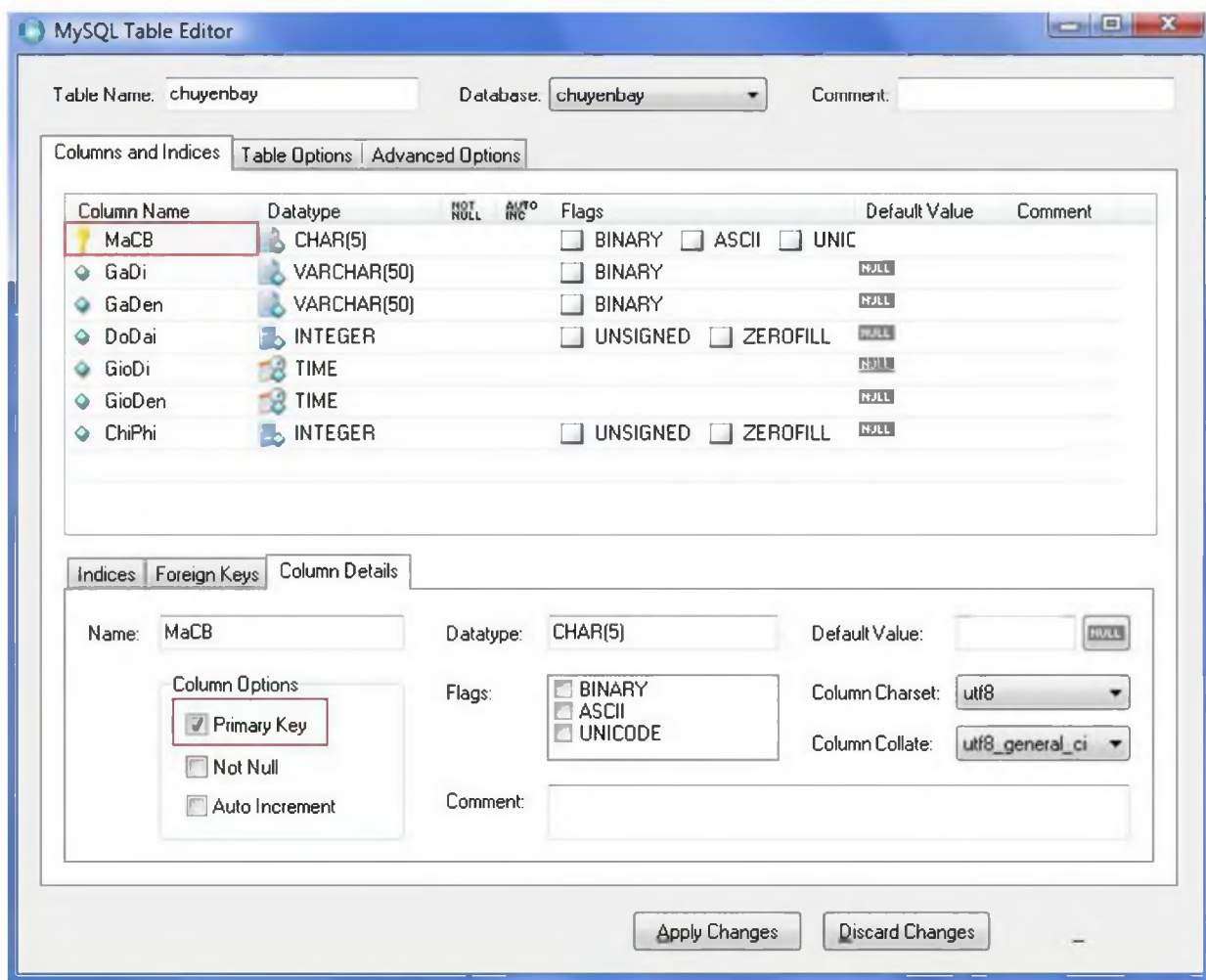
## 8. Tạo khóa chính

+ Khi tạo field đầu tiên trong bảng, MySQL Table Editor mặc định để field này là khóa chính. Tuy nhiên ta có thể chọn field để làm khóa chính như sau:

- + Chọn field làm khóa chính
- + Chọn tab Column Details
- + Trong mục Column Options, check vào Primary Key

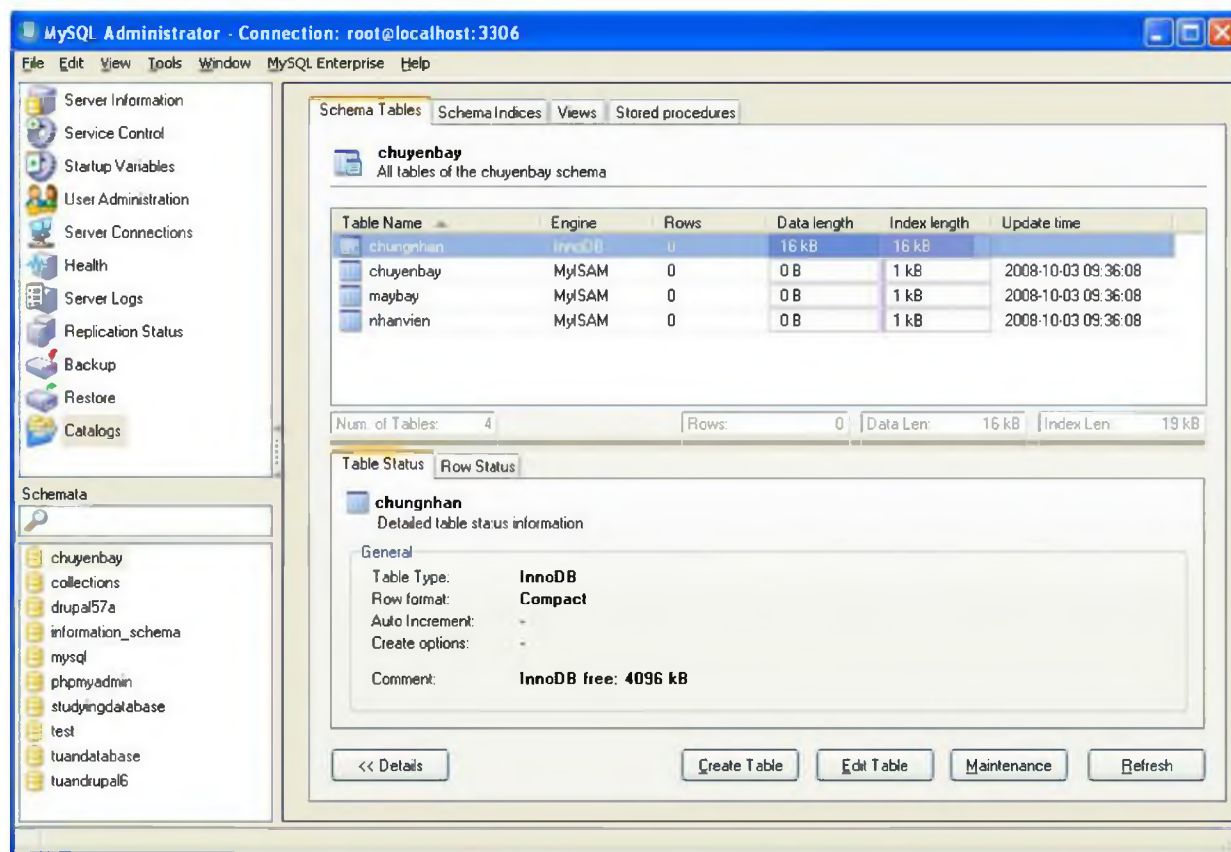


+ Sau check vào Primary Key, ta thấy icon MaCB trong cột Column Name đổi thành hình cái khóa



## 9. Tạo khóa ngoại

+ Chọn bảng cần tạo khóa ngoại bằng cách double click vào tên bảng trong cột Table Name. Ví dụ ta chọn bảng **CHUNGNHAN** trong CSDL **chuyenbay**



+ Trước khi tạo khóa ngoại cho bảng, cần đặt Table Engine của bảng đó và các bảng mà nó tham chiếu tới là **InnoDB**.

**MySQL Table Editor**

Table Name: chungnhan      Database: chuyenbay      Comment: InnoDB free: 4096 kB

Columns and Indices    **Table Options**    Advanced Options

**Storage Engine**

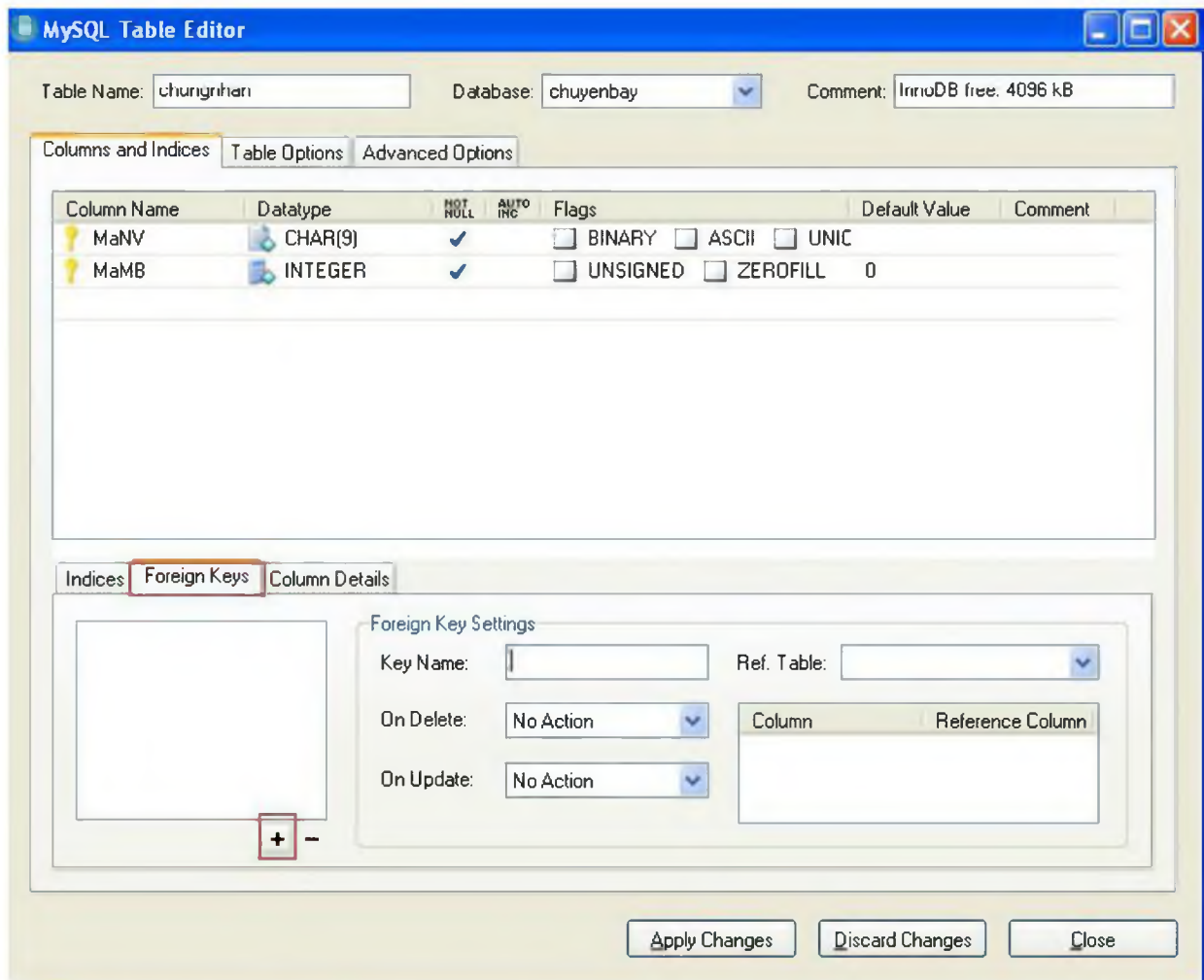
Table Engine: InnoDB  
Supports transactions, row-level locking, and foreign keys

**Character Set**

Charset: utf8      The default character set that is used for the table. Starting from MySQL 4.1 you can specify an individual character set for each column.

Collation: utf8\_general\_ci      Collation method that is used to compare text and sort columns.

Apply Changes    Discard Changes    Close



- + Chọn tab Foreign Keys
- + Click vào dấu cộng như hình trên để tạo mới một khóa ngoại
- + Xuất hiện hộp thoại Add Foreign Key, nhập tên khóa ngoại vào



- + Trong vùng Foreign Key Settings, tại listbox Ref. Table ta chọn bảng tham chiếu tới. Ví dụ ở đây ta chọn bảng tham chiếu là **nhanvien**



Foreign Key Settings

Key Name: chungnhan\_nhanvien\_fk

On Delete: Restrict

On Update: Restrict

Ref. Table: chungnhan  
chuyenbay  
maybay  
nhanvien

Column

Foreign Key Settings

Key Name: chungnhan\_nhanvien\_fk

On Delete: Restrict

On Update: Restrict

Ref. Table: nhanvien

Column	Reference Column
MaNV	MaNV

Apply Changes Discard Changes Close

+ Chọn button Apply Changes

Confirm Table Edit

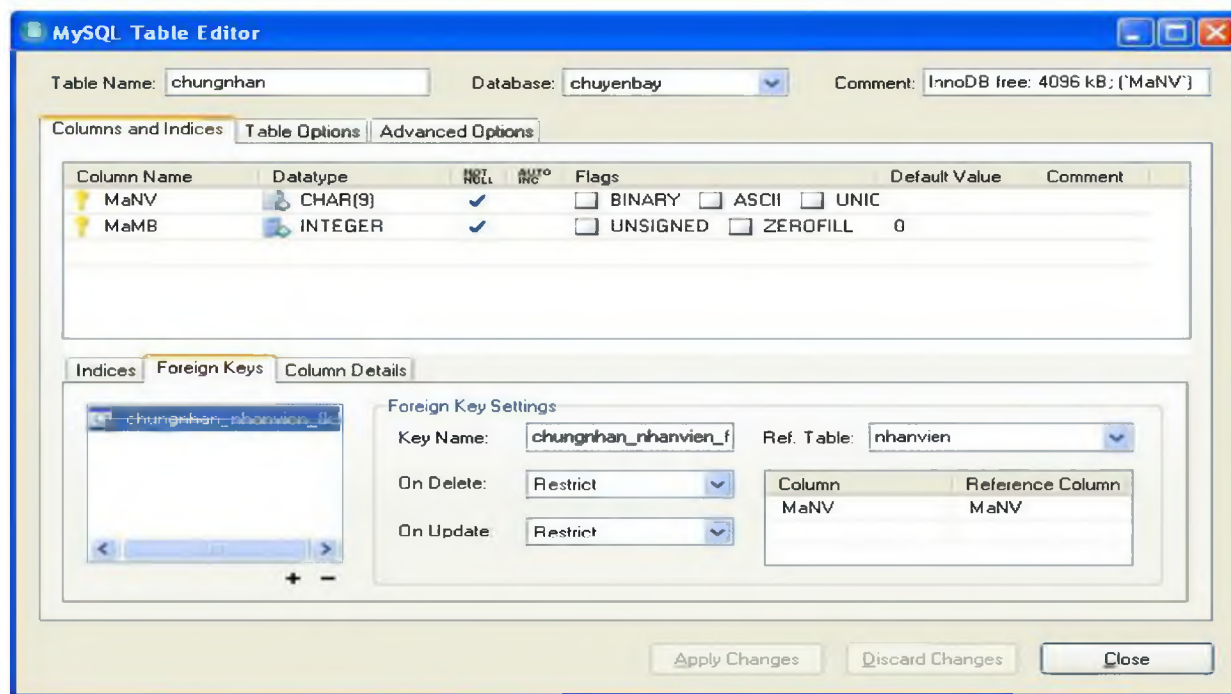
Are you sure you want to execute the following SQL command to apply the changes to the table?

```
ALTER TABLE `chuyenbay`.`chungnhan` DROP FOREIGN KEY `FK_chungnhan_1`;  
ADD CONSTRAINT `chungnhan_nhanvien_fk` FOREIGN KEY `chungnhan_nhanvien_fk`  
(`MaNV`)  
REFERENCES `nhanvien` (`MaNV`)  
ON DELETE RESTRICT  
ON UPDATE RESTRICT  
ROW_FORMAT = DYNAMIC;
```

Execute Cancel

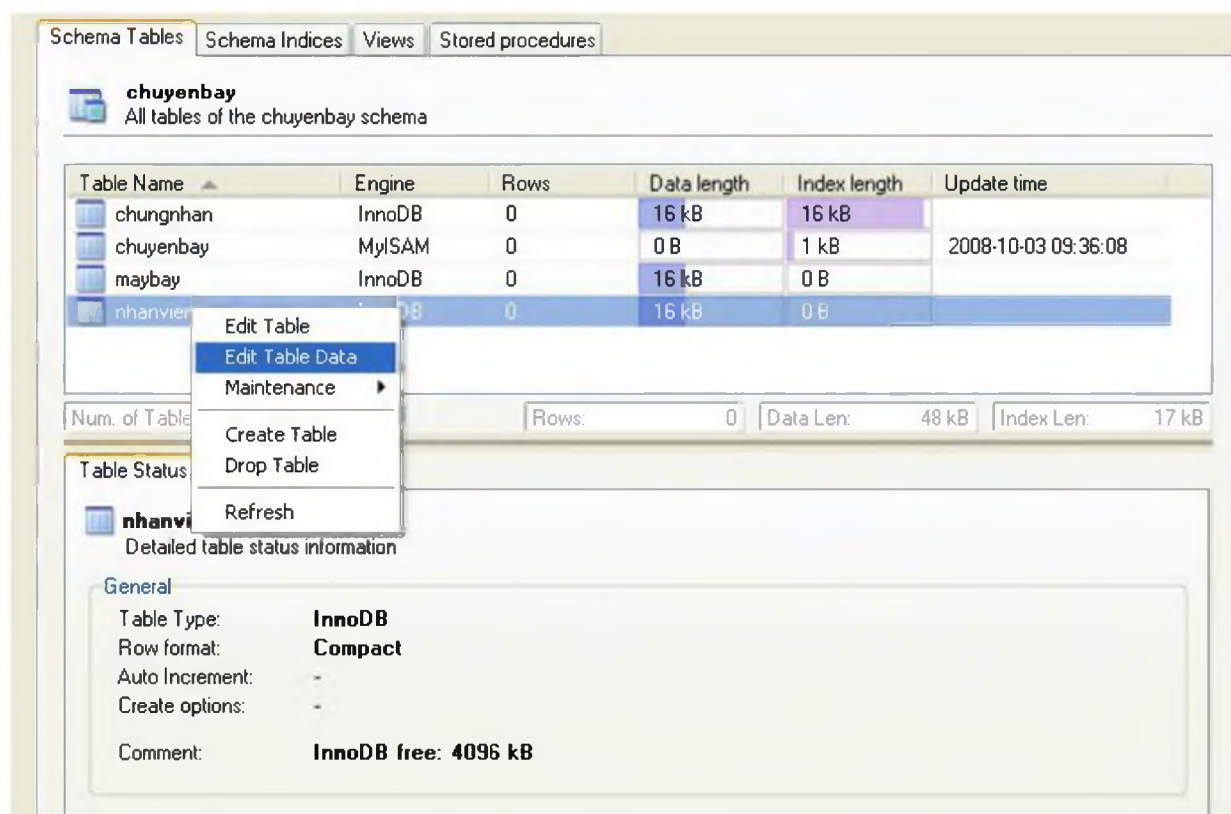
+ Chọn Execute



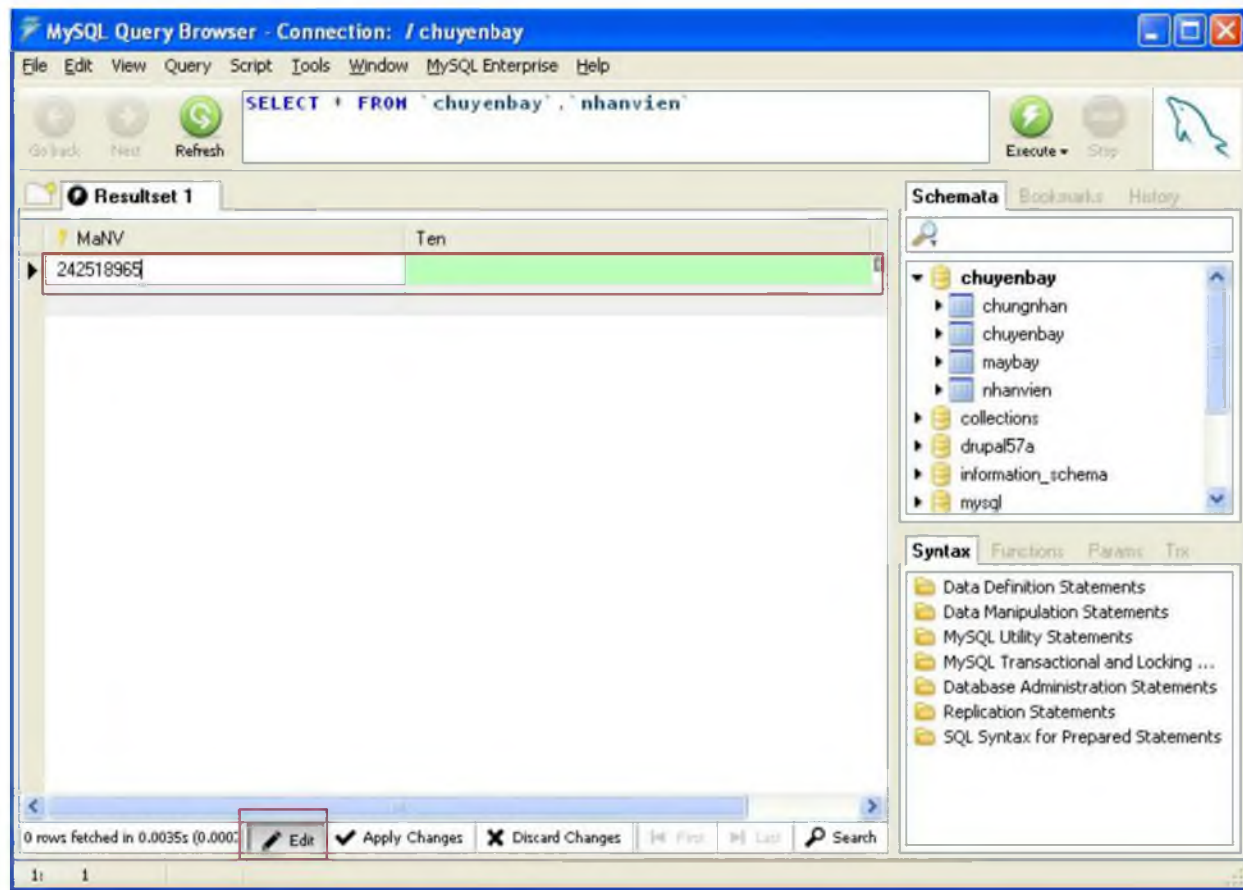


## 10. Insert dữ liệu vào bảng

+ Nhấp chuột phải vào bảng cần insert dữ liệu. Ví dụ ta chọn bảng **nhanvien**



- + Chọn Edit Table Data
- + Xuất hiện hộp thoại MySQL Query Browser



- + Chọn Edit
- + Rồi nhập liệu trong vùng Resultset 1